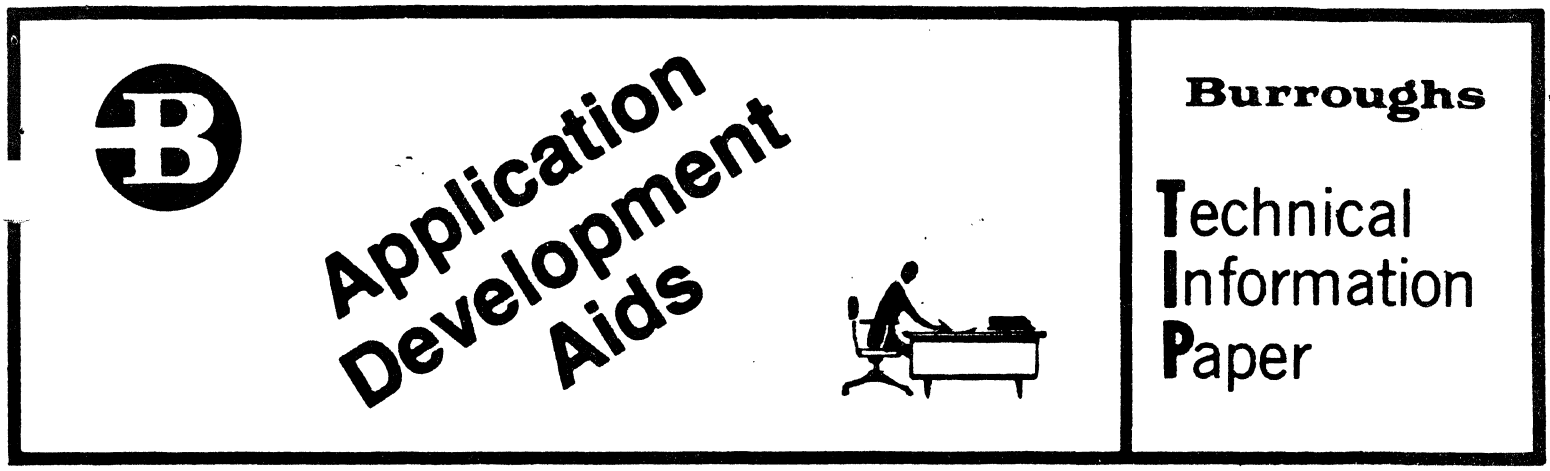

Date: July 12, 1983

Origin: TSC West
El Monte, CA

Product: DMSII

Subject: DMSII DIFFERENCES BETWEEN BURROUGHS SYSTEMS

B5/6/7000 (Large) Systems -- Mark 3.3

B2/3/4000 (Medium) Systems - ASR 6.6

B1000 (Small (Systems) - Mark 11.0

This paper lists the major differences in DMSII between the Large, Medium, and Small Systems implementations. Its intent is to assist the development of compatible DMSII environments across the product line. Upward compatibility is maintained in both DASDL and host language syntax with minor exceptions. Most of even these differences can be resolved by using a compatible subset of features, syntax, or usage. Physical file format compatibility is not maintained across the three implementations.

Copyright © 1983 Burroughs Corporation

TO: BMG Communications Codes DOC 2 and DOC 4

Printed in U.S. of America

1114238-011 (Rev. 2)

+++
+ + +
+ Burroughs + "PRELIMINARY" +
+ Application + Technical +
+ Development + Information +
+ Aids + Paper +
+ + +
+++

Date: March, 1983 Origin: TSC-West
El Monte, Ca.

Product: DMSII

Subject: DMSII DIFFERENCES BETWEEN BURROUGHS SYSTEMS

==========

B5/6/7000 (LARGE) SYSTEMS -- MARK 3.3

B2/3/4000 (MEDIUM) SYSTEMS - ASR 6.6

B1000 (SMALL) SYSTEMS -- MARK 11.0

==========

This paper lists the major differences in DMSII between the Large, Medium, and Small Systems implementations. Its intent is to assist the development of compatible DMSII environments across the product line. Upward compatibility is maintained in both DASDL and host language syntax with minor exceptions. Most of even these differences can be resolved by using a compatible subset of features, syntax, or usage. Physical file format compatibility is not maintained across the three implementations.

GENERAL FEATURES

METHOD OF IMPLEMENTATION

B1000: An interpretive Accessroutines program is bound at run-time with tailored code for SELECT, VERIFY, REQUIRED, REMAPS, etc., and runs as a subsidiary environment of each user program with expanded privileges.

B2/3/4000: Tailored database description and tailored SELECT, VERIFY, etc., code bound with selected interpretive routines generating a handler for each database.

B5/6/7000: Generative. Tailored access routines generated for each database runs at lower lex-level for each user program.

HOST LANGUAGE INTERFACES

B1000: COBOL-74, RPG, COBOL-68.

B2/3/4000: COBOL-74, RPG.

B5/6/7000: COBOL-74, RPG, COBOL-68 ALGOL, PL/I, and APL. An interpretive interface is available to all languages which have access to Libraries (DMINTERPRETER).

ASSIGNMENT OF DATABASE FILES

B1000: Assigned by MCP on a user specified pack or system disk.

B2/3/4000: Assigned by MCP on a user specified (180 byte) pack or (100 byte) disk.

B5/6/7000: Assigned by MCP on a user specified pack or disk, or can be on installation allocated disk or pack.

LEVELS OF SECURITY

B1000: Database, structure, records, and items via remaps and logical databases. Guardfiles may be applied to logical and physical databases. Protection from non-DMSII access may be specified in DASDL.

B2/3/4000: Database, structure, records, and items via remaps and logical databases.

B5/6/7000: Database, structure, records, and items via remaps and logical databases. Guardfiles may be applied to structures and databases and DMSII verbs. Protection from non-DMSII access may be specified in DASDL. Guardfiles may be specified on physical files via the SECURITYGUARD file attribute.

MULTIPLE DATABASES AND CONCURRENT USE

B1000: Maximum of one declaration per program. Maximum of 63 programs having a database open.

B2/3/4000: Maximum of nine databases in use. Maximum of one opened database per program. Maximum of 80 programs having a database open.

B5/6/7000: There are no restrictions on the number of databases that can remain open concurrently, nor the number of databases that can be open per program (except RPG). However, multiple update opens are logically not recoverable. Maximum of 192 programs using a single database.

OPTIONS

B1000: AUDIT, STATISTICS, KEYCOMPARE. Addresscheck is included (not options)and apply only to sets. Readahead and writeahead are automatic.

B2/3/4000: AUDIT, STATISTICS, KEYCOMPARE. Addresscheck is included (not an option). Readahead via buffer specs; writeahead automatic (but resetable); checksum default is ON.

B5/6/7000: AUDIT, STATISTICS, KEYCOMPARE, ADDRESSCHECK, BACKOUTTOSPT, RESTART TRANSACTIONS ONLY, TPSDUALUPDATE, RDSSTORE. Readahead via buffer specs; writehead automatic; checksum default is OFF; reblocking and modeling may be specified.

PARAMETERS

B1000: SYNCPOINT, CONTROLPOINT, MAXWAIT.

B2/3/4000: SYNCPOINT, CONTROLPOINT, ALLOWEDCORE, STACKSIZE.

B5/6/7000: SYNCPOINT, CONTROLPOINT, ALLOWEDCORE.

GLOBAL DATA

B1000: Not implemented.

B2/3/4000: Not implemented.

B5/6/7000: Can contain data items or POPULATION items.

DATA SET ORGANIZATIONS

B1000: STANDARD for disjoint; UNORDERED and ORDERED for embedded. Spanning sets are allowed only on the disjoint level.

B2/3/4000: STANDARD, DIRECT, UNORDERED, and RANDOM for disjoint; STANDARD and UNORDERED for embedded.

B5/6/7000: STANDARD, DIRECT, UNORDERED, RANDOM, ORDERED, or COMPACT data sets for disjoint; STANDARD, ORDERED, UNORDERED or COMPACT for embedded data sets. Sets may span any type of data set except ORDERED.

VARIABLE FORMAT RECORDS

B1000: Maximum of 255 format types can be declared in each variable format data set. Record type may be ALPHA or NUMBER.

B2/3/4000: Not implemented.

B5/6/7000: Maximum of 254 format types can be declared in each variable format data set. Record type is restricted to unsigned NUMBER.

DATA TYPES

B1000: ALPHA, NUMBER, GROUP, and RECORD TYPE.

B2/3/4000: ALPHA, NUMBER, GROUP, BOOLEAN, FIELD.

B5/6/7000: ALPHA, NUMBER, GROUP, BOOLEAN, FIELD, REAL, also special purpose types of COUNT, POPULATION, RECORD TYPE, LINK, FILLER, and VIRTUAL (remaps only).

INITIAL VALUES ON DATA ITEMS ON CREATE

B1000 and B2/3/4000:
Defaults to all bits on (NULL value) or can be user defined INITIALVALUEs rather than default.

B5/6/7000: Default depends upon data item type and whether the item is an occuring item. Defaults may be set to user defined values. User can also specify INITIALVALUEs rather than default.

SET AND AUTOMATIC SUBSET ORGANIZATIONS

B1000: Index sequential, Index random.

B2/3/4000: Index sequential, Ordered list, Unordered list.

B5/6/7000: Index sequential, Ordered list, Unordered list, and, for disjoint only, Index random, and bit vector.

SETS (ORDERING OR RETRIEVAL STRUCTURES)

B1000: Any number at disjoint level for disjoint data sets. Only one (ACCESS) can span each ordered data set.

B2/3/4000 and B5/6/7000:
Any number at disjoint level or embedded level. Embedded sets may span embedded data sets at the same level and are required for STANDARD and COMPACT data sets.

AUTOMATIC SUBSETS

B1000: Any number allowed at disjoint level. Not implemented to span embedded structures.

B2/3/4000 and B5/6/7000:
Automatic subsets allowed for disjoint as well as for embedded data sets.

MANUAL SUBSETS

B1000: Must be embedded and refer to disjoint data set.

B2/3/4000 and B5/6/7000:
Can be disjoint or embedded. Embedded subsets may refer to either disjoint or embedded data set but must satisfy scope rules.

MANUAL SUBSET ORGANIZATIONS

B1000: Ordered list, unordered list.

B2/3/4000: Ordered list, unordered list, index sequential.

B5/6/7000: Ordered list, unordered list, index sequential, and, for disjoint only, index random and bit vector.

LINKS

B1000: Unordered manual subset with one entry have the physical pointers in the master record, similar to unprotected link.

B2/3/4000: Unordered manual (embedded) subset with one or two entries have the physical pointers in the master record, similar to unprotected link.

B5/6/7000: UNPROTECTED, COUNTED, VERIFIED, SYMBOLIC, and SELFCORRECTING links are implemented.

KEY DATA (DATA IN INDEX TABLE)

B1000 and B2/3/4000:
Not implemented.

B5/6/7000: Can be a collection of data items declared in the data set.

PARTITIONED DATA SETS AND SETS

B1000 and B2/3/4000:
Not implemented.

B5/6/7000: Only embedded structures can be partitioned. The partitioned master record must be the record of a disjoint data set.

PRIME SETS

B1000: Deimplemented.

B2/3/4000: Not implemented.

B5/6/7000: Used at reorganization for sequencing data sets.

MAXENTRIES FOR SETS

B1000: MAXENTRIES for Index Sequential Sets represents the total number of entries to be handled at the Fine Table level.

B2/3/4000: MAXENTRIES for Index Sequential Sets represents the total number of entries in ALL tables of the Set: Root, Coarse and Fine.

B5/6/7000: MAXENTRIES/POPULATION for Index Sequential Sets represents the total number of entries to be handled at the Fine Table level.

PHYSICAL ATTRIBUTES COMPATIBILITY

The following tables show the preferred choice for compatible syntax.

FOR DATA SETS:

B1000	B2/3/4000	B5/6/7000
N/A	ALLOCATE AT END	ALLOCATE AT END
AREALENGTH in BLOCKS	AREALENGTH in BLOCKS	AREALENGTH in BLOCKS
BLOCKSIZE in RECORDS	BLOCKSIZE in RECORDS	BLOCKSIZE in RECORDS
N/A	MODULUS	MODULUS

FOR SET AND SUBSETS:

B1000	B2/3/4000	B5/6/7000
AREALENGTH in BLOCKS	AREALENGTH in BLOCKS	AREALENGTH in TABLES
BLOCKSIZE in ENTRIES	BLOCKSIZE in ENTRIES	BLOCKSIZE (default units is entries)
MODULUS	N/A	MODULUS
SPLITFACTOR	LOADFACTOR	LOADFACTOR or SPLITFACTOR

FOR ALL STRUCTURES:

B1000	B2/3/4000	B5/6/7000
AREAS < 100 (105 max)	AREAS < 100 (100 max)	AREAS < 100 (1K max)
FAMILYNAME	FAMILYNAME (6 chars)	DISKPACK
N/A	CHECKSUM	CHECKSUM
N/A	BUFFERS = x + y + z	BUFFERS = x + y + z

HOST LANGUAGE INTERFACE

OPEN TYPES

B1000 and B2/3/4000:
UPDATE, INQUIRY.

B5/6/7000: UPDATE, INQUIRY, TRUPDATE.

HOST LANGUAGE VERBS

B1000: LOCK replaces MODIFY in COBOL-74. SET not implemented.

B2/3/4000: LOCK not implemented on ASR 6.6, scheduled for 6.7.

B5/6/7000: ASSIGN, GENERATE,MID-TRANSACTION, DMTERMINATE; COMPUTE (COBOL); PUT, GET, DMTEST (ALGOL) are additional verbs.

SELECTION EXPRESSIONS

B1000: Partial and complex selection expressions allowed on sets or subsets in COBOL-74 and RPG. Complex selection expressions only in COBOL.

B2/3/4000 and B5/6/7000:
Partial and complex selection expressions allowed on sets or subsets.

FIND KEY OF

B1000: Not implemented.

B2/3/4000: Not implemented on ASR 6.6, scheduled for 6.7.

B5/6/7000: Implemented for selection expressions using sets or subsets.

DMERROR DECLARATIVES IN COBOL

B1000: Not Implemented.

B2/3/4000 and B5/6/7000:
Implemented.

EXCEPTION HANDLING

DMSTATUS

B1000: Returns DMERROR and exception type.

B2/3/4000 and B5/6/7000:
Returns DMERROR, exception type, exception subtype and structure number.

DM EXCEPTION INCOMPATIBILITY

B1000: KEYCHANGED exception possible if duplicates are not allowed on disjoint SET or if key items are changed for an ORDERED DATA SET.

B2/3/4000: KEYCHANGED exception possible if duplicates are not allowed on a RANDOM data set. SECURITYERROR not implemented.

B5/6/7000: KEYCHANGED exception possible if duplicates are not allowed on embedded set.

ON EXCEPTION CLAUSE

B1000: Conditional statements are allowed in the ON EXCEPTION clause for COBOL verbs.

B2/3/4000: Only imperative statements are allowed in the ON EXCEPTION clause for COBOL verbs. Conditional statements will be allowed on ASR 6.7.

B5/6/7000: Conditional statements are allowed in the ON EXCEPTION clause.

HANDLING OF CURRENT RECORD/PATH ON NOTFOUND EXCEPTION

B1000: COBOL-68 will update the current path and current record to the next entry beyond where the selection expression "would have been satisfied" on an equal condition for index-sequential sets only. COBOL-74 and RPG will not update "currents" after a NOTFOUND exception.

B2/3/4000: "Currents" remain unchanged on any NOTFOUND exception, however, a (ASR 6.7) COBOL option, SAVECP, will cause the to be updated as in B1000 COBOL-68.

B5/6/7000: "Currents" are always updated on NOTFOUND exception.

AUDIT AND RECOVERY

DUPLICATE AUDIT FILE

B1000: Not implemented, however, audit may be assigned to tape.

B2/3/4000: Allowed to pack or disk, no tape audit.

B5/6/7000: Primary and secondary can be on disk, pack, or tape.

RECONSTRUCTION OF DATABASE FILE PORTIONS

B1000 and B2/3/4000:
Can be done only when database is closed. Scope is the entire database or selected files.

B5/6/7000: Can be done while applications process against database. Reconstruction can be done down to a row level in a file.

DUMPING OF DATABASE

B1000: Standard library maintenance on inquiry or closed database.

B2/3/4000: Standard library maintenance on closed database for ASR 6.6. On-line dump scheduled for 6.7.

B5/6/7000: Special utility allows dumping of an open database.

ROLLBACK

B1000: Not implemented.

B2/3/4000 and B5/6/7000:
Implemented, but database not available during process.

CAPTURING RESTART INFORMATION

B1000 and B2/3/4000:
The restart information is stored either at BEGIN-TRANSACTION or at END-TRANSACTION, by the DMSII effecting a STORE of the Restart Record Area. The record will be STOREd according to the setting of the Current Record Pointer for the Restart Data Set; if it points at the previously STOREd Restart Record, it will overwrite that record; if the path has been changed with a FIND, MODIFY, etc., the Restart information will be written to

that record; if a (RE)CREATE has been done, then a new record will be stored. In other words, its actions are just like a Standard Data Set.

B5/6/7000: The restart information is captured only in the Audit Trail at BEGIN or END-TRANSACTION. ABORT and H/L Recovery, REBUILD and ROLLBACK will retrieve all the appropriate Restart Areas and put them into the RESTART DATA SET during the Recovery process. Hence, a program can have only one Restart Area, its latest on the Audit Trail. The RDSSTORE option causes an implementation similar to B1000 and B2/3/4000 Systems.

RESTART RECORD HANDLING

B1000 and B2/3/4000:
To remove all vestiges of Restart Records in the Data Base, a program may execute the following sequence of Host statements :

```
BEGIN-TRANSACTION RDS NO-AUDIT.
DELETE RDS.
END-TRANSACTION RDS NO-AUDIT SYNC.
```

where the current path for RDS is to the program Restart Record. If any Recovery (H/L or ABORT) were to subsequently occur, this program would no longer have a Restart Record. It should be noted that if a program relies on the presence of a Restart record to indicate whether restart is required, and it performs this immediately before the CLOSE of the Data Base, then there remains a window between the END-TRANSACTION and the CLOSE such that if the Recovery situation is invoked there is an ambiguity in the Restart Logic as to the beginning (creation of the restart record) or the ending (deletion of the restart record).

B5/6/7000: Since the Restart Area is only stored in the RESTART DATA SET by Recovery functions, it can only be DELETEd after a Recovery function has been invoked (of course, records STOREd in the normal fashion may be DELETEd normally). In addition, execution of the sequence :

```
BEGIN-TRANSACTION RDS NO-AUDIT.
END-TRANSACTION RDS NO-AUDIT SYNC.
```

will result in the Restart Area from the PREVIOUS Transaction remaining valid. The only way a program can guarantee to have no Restart Areas (apart from always doing '... NO-AUDIT') is to actually close the Data Base; for all cases of a program with the Data Base OPEN UPDATE, the latest Restart Area for a completed transaction on the Audit Trail will be returned. The RDSSTORE option causes an implementation similar to B1000 and B2/3/4000 Systems.

DEFAULTS FOR AUDIT <RESTART DS> (store)

B1000: Default AUDIT at BEGIN-TRANSACTION for COBOL-68, at END-TRANSACTION for RPG, and no default for COBOL-74.

B2/3/4000: Default AUDIT at END-TRANSACTION.

B5/6/7000: No default AUDIT.

REORGANIZATION

B1000 and B2/3/4000:
There is no restrictions on the number of disjoint structures that may be reorganized or garbage collected in a single compilation. The database must be closed during reorganization.

B5/6/7000: Only one disjoint structure (or disjoint cluster) can be reorganized per DASDL, BUILDREORG, and REORG executions. The accessroutines and utilities, etc., compilations may be delayed until all reorganizations are complete. It is possible that much reorganization process may be done while the database is in use.

MISCELLANEOUS FEATURES AND UTILITIES

DATA DICTIONARY

B1000 and B2/3/4000:
Not implemented.

B5/6/7000: Implemented.

TRANSACTION PROCESSING SYSTEM (TPS)

B1000 and B2/3/4000:
Not implemented.

B5/6/7000: Implemented.

INQUIRY

B1000 and B2/3/4000:
General purpose program for all data bases. Special file generated for each database for the inquiry interface.

B5/6/7000: Special inquiry program generated for each database.

OTHER UTILITIES

B1000: Dictionary decompiler; analyzers for dictionary structures, audit trail, and database.

B2/3/4000: Analyzers for database, control file, DDF, and DBP dumps; verify and print audit trail; report statistics; control recovery and reorganization.

B5/6/7000: UTILITY for database load and dump (backup), recovery, initialize, and data/control information printing; data load and dump; audit trail analyzer; audit trail verify and copy; directory dump; database analyzer; monitoring; certification; and database parameter control.

LIMITS AND MAXIMUM VALUES

The following table contains certain pertinent features and physical attributes of DMSII and shows the maximum values and limits where applicable. Blanks indicate that the limit, if one exists, is developed by machine or operating systems constraints, or a practical limit will manifest itself before hitting the theoretical limit. "N/A" indicates the absent feature.

	B1000	B2/3/4000	B5/6/7000
Identifier size	17 chars	17 chars	17 chars
ALPHA items	8191 chars	10,000 chars	4095 chars
ALPHA items as keys	511 chars	100 chars	4095 chars
NUMBER items	23 digits	23 digits	23 digits
NUMBER items as keys	12 digits	12 digits	12 digits
FIELD items	N/A	48 (bits)	48 (bits)
"Comments" size	172 chars	255 chars	255 chars
OCCURS (n) TIMES	1023	1023	1023
Subscript levels	3	49	
DATA SET record size	8192 bytes	20,000 bytes	24,570 bytes
BLOCKSIZE (data set)	127 records	999 records	
TABLESIZE (set)	8100 bytes	9999 entries or 20K bytes	780K bytes
AREALENGTH (sectors)	65,535	99,999,999	1,048,575
MAXRECORDS	123,863,040	99,999,999	268,435,456
AREAS	105	100	1000
Index sequential table levels	15	99	22
Database structures	255	256 (ASR 6.6) 512 (ASR 6.7)	1023
Structures invoked per program			256

Table of Contents

GENERAL FEATURES. 1
METHOD OF IMPLEMENTATION. 1
HOST LANGUAGE INTERFACES. 1
ASSIGNMENT OF DATABASE FILES. 1
LEVELS OF SECURITY. 1
MULTIPLE DATABASES AND CONCURRENT USE 2
DASDL 3
OPTIONS 3
PARAMETERS. 3
GLOBAL DATA 3
DATA SET ORGANIZATIONS. 3
VARIABLE FORMAT RECORDS 4
DATA TYPES. 4
INITIAL VALUES ON DATA ITEMS ON CREATE. 4
SET AND AUTOMATIC SUBSET ORGANIZATIONS. 4
SETS (ORDERING OR RETRIEVAL STRUCTURES) 4
AUTOMATIC SUBSETS 5
MANUAL SUBSETS. 5
MANUAL SUBSET ORGANIZATIONS 5
LINKS 5
KEY DATA (DATA IN INDEX TABLE). 6
PARTITIONED DATA SETS AND SETS. 6
PRIME SETS. 6
MAXENTRIES FOR SETS 6
PHYSICAL ATTRIBUTES COMPATIBILITY 7
HOST LANGUAGE INTERFACE 8
OPEN TYPES. 8
HOST LANGUAGE VERBS 8
SELECTION EXPRESSIONS 8
FIND KEY OF 8
DMERROR DECLARATIVES IN COBOL 8
EXCEPTION HANDLING. 9
DMSTATUS. 9
DM EXCEPTION INCOMPATIBILITY. 9
ON EXCEPTION CLAUSE 9
HANDLING OF CURRENT RECORD/PATH ON NOTFOUND EXCEPTION . . 9
AUDIT AND RECOVERY. 11
DUPLICATE AUDIT FILE. 11
RECONSTRUCTION OF DATABASE FILE PORTIONS. 11
DUMPING OF DATABASE 11
ROLLBACK. 11
CAPTURING RESTART INFORMATION 11
RESTART RECORD HANDLING 12
DEFAULTS FOR AUDIT <RESTART DS> (store) 13
REORGANIZATION. 13
MISCELLANEOUS FEATURES AND UTILITIES. 14
DATA DICTIONARY 14
TRANSACTION PROCESSING SYSTEM (TPS) 14
INQUIRY 14
OTHER UTILITIES 14
LIMITS AND MAXIMUM VALUES 15